

AD-A049 917 OFFICE OF THE PROJECT MANAGER SELECTED AMMUNITION DO--ETC F/G 5/10
A COMPUTER PROGRAM FOR SOLVING LEARNING CURVE PROBLEMS. (U)
DEC 77 L M SMITH

UNCLASSIFIED

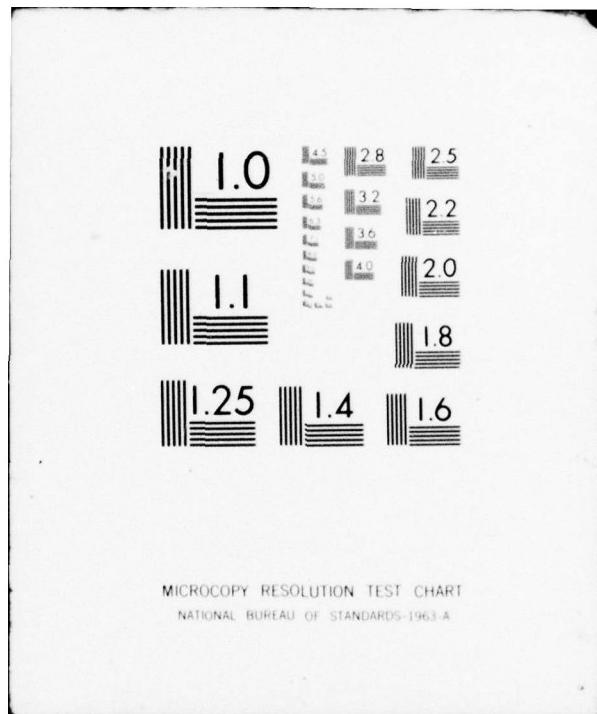
PMSA-2-6

NL

| OF |
AD
A049917



END
DATE
FILED
3 -78
DDC



AU NO. _____
DDC FILE COPY

ADA 049917



REPORT NO. PMSA-2-6

A COMPUTER PROGRAM FOR
SOLVING LEARNING CURVE PROBLEMS

DECEMBER 1977

LOUIS M. SMITH

PROJECT MANAGER
SELECTED AMMUNITION
DOVER, NEW JERSEY

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

D D C
REPORT TO
FEB 14 1978
R E S U L T S
D

Q

| | |
|---------------------------------|---|
| ACCESSION for | |
| RTB | White Section <input checked="" type="checkbox"/> |
| DDC | Buff Section <input type="checkbox"/> |
| UNANNOUNCED | |
| JUSTIFICATION | |
| BY | |
| DISTRIBUTION/AVAILABILITY CODES | |
| BEST. AVAIL. AND OR SPECIAL | |
| A | |

(14) REPORT NO. PMSA-2-6

(6) A COMPUTER PROGRAM FOR
SOLVING LEARNING CURVE PROBLEMS.

(9) Final rept.

(10) LOUIS M. SMITH
Operations Research
Analyst

(11) Dec 77

(12) 42P.

D D C
RECORDED
FEB 14 1978
RECORDED

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

389 515 JOB

TABLE OF CONTENTS

| | Page |
|---|-------------|
| ABSTRACT | 1 |
| INTRODUCTION | 1 |
| PROGRAM INPUT RESTRICTIONS | 3 |
| PROGRAM DESCRIPTION | 4 |
| DESCRIPTION ON PROGRAM ENTRY | 4 |
| DETAILED DESCRIPTION | 4 |
| TYPICAL PROBLEMS: INPUT AND OUTPUT | 9 |
| Subroutine MIDPOINT | 9 |
| Subroutine FUNTOT | 11 |
| Subroutine FUNAV | 12 |
| Subroutine TOTCOST/Subroutine WHATIF | 14 |
| Subroutine MIDBEST | 17 |
| Subroutine LOTSIZE | 20 |
| Subroutine AVUCOST | 21 |
| PROGRAM LISTING | 22a |

ABSTRACT

This report describes a computer program for solving problems dealing with learning curve theory. The program is interactive, written for the remote computer terminal. It is written in Fortran IV language for the CDC 6500/6600 computer system at USA Research and Development Command, Dover, New Jersey but can be easily adapted to other computer systems.

INTRODUCTION

The history of experience curve theory dates to about 1936 and there are hosts of references describing the theory and its application. This report assumes a knowledge by the potential user of both 'unit theory' experience curve manipulations and remote computer terminal usage.

The elementary experience (learning) curve theory postulates that as the numbers of an item produced increases sequentially, the cost of the item decreases in a relationship defined by the power curve:

$$Y = AX^B$$

Where: X = Item number

Y = Cost of item X

A = Cost of first unit (X = 1)

B = Learning Slope Exponent

Expressed logarithmically the power curve assumes the following form:

$$\log Y = \log A + B \cdot \log X$$

This form of the power curve plots a straight line on log/log graph paper and it is this relationship which is exploited in the computer program to provide solutions to typical learning curve problems.

This report will describe the computer program usage by presenting typical problems for each of the eight computational segments.

All user input to the remote terminal will be underlined to acquaint the user with the kind of information that is expected by each of the program segments.

The program is written in Fortran IV for use by the CDC 6500/6600 at ARRADCOM, Dover, NJ but can be adapted to any computer system.

PROGRAM INPUT RESTRICTIONS

Input formats are relatively uniform throughout each segment of the program and are few enough to commit to memory.

The restrictions are:

(1) All lot size input must contain 9 digits, right-hand oriented. Thus, unit number 1 of a lot is input as 000000001; unit number 15560 is input as 000015560. It is obvious, therefore, that the program is capable of handling very large lot sizes.

(2) 'Number of lots' input must contain 2 digits right-hand oriented. One (1) lot is input as 01. The maximum number of lots is 15. EXCEPTION: Program segment WHATIF will accept only 5 lots max. for its computations and only one (1) digit is required for input.

(3) All cost data must contain a decimal point. The program will accept ten digits max. (with the decimal point considered a digit), including costs exact to the penny. Maximum acceptable cost, then, may have the following appearance:

99999999.00 OR

999999999.

(4) The learning slope exponent is input as it appears in learning slope tables preceded by the minus (-) sign.

PROGRAM DESCRIPTION

Description on Program Entry

When the program is entered the user may proceed directly to a desired segment, depending upon his familiarity with the segment code letters, or he may have a short description and the code letter associated with each segment, Figure 1.

FIGURE 1

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

Y

THIS PROGRAM CONSISTS OF THE FOLLOWING SEGMENTS:

- A. MIDPOINT - DETERMINES LOT MIDPOINTS
- B. FUNTOT - DETERMINES FIRST UNIT COST
- C. FUNAV - DETERMINES FIRST UNIT COST
- D. TOTCOST - DETERMINES TOTAL COST OF LOT
- E. AVUCOST - DETERMINES ITEM AVERAGE COST
- F. MIDBEST - DETERMINES FIRST UNIT COST AND LEARNING SLOPE EXPONENT
- G. WHATIF - DETERMINES LOT SIZES
- H. LOTSIZE - DETERMINES SIZE OF A SINGLE LOT

IF YOU NEED MORE DETAIL TYPE LETTER OR TYPE NO

In response to the last direction, input of 'NO' will allow the user to enter any of the computation segments of the program, Figure 2.

Detailed Description

Additional description of any program segment is provided by an input of any letter from A to H. All descriptions, obtained by responding 'NO' to each option, is provided in Figure 3. Note that each description is

followed with the options that allow the user to enter the computational segment or call for additional descriptions. Errors in input creates an error message and the opportunity to re-enter a corrected input.

FIGURE 2

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

Y

THIS PROGRAM CONSISTS OF THE FOLLOWING SEGMENTS:

- A. MIDPOINT - DETERMINES LOT MIDPOINTS
- B. FUNTOT - DETERMINES FIRST UNIT COST
- C. FUNAV - DETERMINES FIRST UNIT COST
- D. TOTCOST - DETERMINES TOTAL COST OF LOT
- E. AVUCOST - DETERMINES ITEM AVERAGE COST
- F. MIDBEST - DETERMINES FIRST UNIT COST
AND LEARNING SLOPE EXPONENT
- G. WHATIF - DETERMINES LOT SIZES
- H. LOTSIZE - DETERMINES SIZE OF A SINGLE LOT

IF YOU NEED MORE DETAIL TYPE LETTER OR TYPE NO
NO

TYPE IN LETTER OF PROGRAM YOU DESIRE
A

THIS IS MIDPOINT

ENTER LEARNING SLOPE EXPONENT

FIGURE 3

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

Y

THIS PROGRAM CONSISTS OF THE FOLLOWING SEGMENTS:

- A. MIDPOINT - DETERMINES LOT MIDPOINTS
- B. FUNTOT - DETERMINES FIRST UNIT COST
- C. FUNAV - DETERMINES FIRST UNIT COST
- D. TOTCOST - DETERMINES TOTAL COST OF LOT
- E. AVUCOST - DETERMINES ITEM AVERAGE COST
- F. MIDBEST - DETERMINES FIRST UNIT COST
AND LEARNING SLOPE EXPONENT
- G. WHATIF - DETERMINES LOT SIZES
- H. LOTSIZE - DETERMINES SIZE OF A SINGLE LOT

IF YOU NEED MORE DETAIL TYPE LETTER OR TYPE NO

A

PROGRAM MIDPOINT

THIS SEGMENT CALCULATES LOT MIDPOINTS IF THE LEARNING SLOPE EXPONENT AND LOT QUANTITIES (FIRST AND LAST ITEMS) ARE KNOWN. TYPE A TO CALL PROGRAM OR TYPE NO

NO

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

B

PROGRAM FUNTOT

THIS SEGMENT CALCULATES FIRST UNIT COST IF THE LEARNING SLOPE EXPONENT, LOT QUANTITIES (FIRST AND LAST ITEMS) AND TOTAL COST OF THE LOT ARE KNOWN. TYPE B TO CALL THIS PROGRAM OR TYPE NO

NO

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

C

PROGRAM FUNAV

THIS SEGMENT CALCULATES FIRST UNIT COST IF THE LEARNING SLOPE EXPONENT, LOT QUANTITIES (FIRST AND LAST ITEMS) AND THE AVERAGE UNIT COST OF AN ITEM IN THE LOT ARE KNOWN. TYPE C TO CALL THIS PROGRAM OR TYPE NO

NO

FIGURE 3 (CONTINUED)

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

D

PROGRAM TOTCOST

THIS SEGMENT CALCULATES THE TOTAL COST OF A LOT IF THE LEARNING SLOPE EXPONENT, FIRST UNIT COST, AND LOT QUANTITIES (FIRST AND LAST ITEMS), ARE KNOWN. TYPE D TO CALL THIS PROGRAM OR TYPE NO

NO

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

E

PROGRAM AVUCOST

THIS SEGMENT CALCULATES THE AVERAGE COST OF AN ITEM IN A LOT IF THE LEARNING SLOPE EXPONENT, FIRST UNIT COST AND LOT QUANTITIES (FIRST AND LAST ITEMS), ARE KNOWN. TYPE E TO CALL THIS PROGRAM OR TYPE NO

NO

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

F

PROGRAM MIDBEST

THIS SEGMENT CALCULATES FIRST UNIT COST AND LEARNING SLOPE EXPONENT FROM LOT DATA IF THE AVERAGE UNIT COST OF AN ITEM IN EACH OF THE LOTS AND LOT QUANTITIES (FIRST AND LAST ITEMS) ARE KNOWN. LEAST MEAN SQUARE FIT OF DATA IS UTILIZED. STANDARD ERROR OF ESTIMATE AND CORRELATION COEFFICIENT ARE CALCULATED. PROGRAM CAN ACCOMODATE MAXIMUM OF NINE(9) LOTS. TYPE F TO CALL THIS PROGRAM OR TYPE NO.

NO

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

G

PROGRAM WHATIF

THIS SEGMENT COMPUTES THE SIZE OF 1 OR MORE LOTS(5 MAX), IF THE TOTAL COST OF THE LOT(S), THE 1ST UNIT COST, AND THE LEARNING SLOPE EXPONENT ARE KNOWN. TYPE G TO CALL THIS PROGRAM OR TYPE NO

NO

FIGURE 3 (CONTINUED)

WHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER

H

PROGRAM LOTSIZE

THIS SEGMENT COMPUTES THE SIZE OF A SINGLE LOT IF THE TOTAL COST OF THE LOT, THE 1ST UNIT COST, THE LEARNING SLOPE EXPONENT AND THE THE FIRST UNIT OF THE LOT ARE KNOWN. IT DIFFERS FROM SEGMENT WHATIF IN THAT THE FIRST UNIT OF LOTSIZE CAN BE ANY UNIT (OTHER THAN UNIT NO. 1). TYPE H TO CALL THIS PROGRAM OR TYPE NO

TYPICAL PROBLEMS: INPUT AND OUTPUT

Subroutine MID-POINT

Problem: (a) What are the lot mid-points of the following lot procurements of an item whose production learning slope is 85% (-.234465)? (b) If lots 2/3 and 4/5 are combined what are the new lot mid-points of the three lots if the learning slope is 91% (-.136061)?

| | <u>Lot</u> | <u>First Unit</u> | <u>Last Unit</u> |
|-----|------------|-------------------|------------------|
| (a) | 1 | 1 | 750 |
| | 2 | 751 | 2000 |
| | 3 | 2001 | 5000 |
| | 4 | 5001 | 10000 |
| | 5 | 10001 | 25000 |
| (b) | 1 | 1 | 750 |
| | 2 | 751 | 5000 |
| | 3 | 5001 | 25000 |

The program output is shown in Figure 4.

FIGURE 4

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

A

THIS IS MIDPOINT

ENTER LEARNING SLOPE EXPONENT
-.234465

FIGURE 4 (CONTINUED)

HOW MANY LOTS
05

ENTER FIRST UNIT OF EACH LOT
0000000001
0000000751
0000002001
0000005001
000010001

ENTER LAST UNIT OF EACH LOT
0000000750
0000002000
0000005000
000010000
000025000

DATA FOR SLOPE OF 85.00 PCT.

| FIRST UNIT OF LOT | LAST UNIT OF LOT | LOT MID-POINT |
|-------------------|------------------|---------------|
| 1 | 750 | 243 |
| 751 | 2000 | 1314 |
| 2001 | 5000 | 3362 |
| 5001 | 10000 | 7324 |
| 10001 | 25000 | 16808 |

MORE DATA FOR MIDPOINT PROGRAM - ANS. YES OR NO
YES

SAME LEARNING SLOPE EXPONENT - ANS. YES OR NO
NO

ENTER LEARNING SLOPE EXPONENT
-136061

HOW MANY LOTS
03

FIGURE 4 (CONTINUED)

ENTER FIRST UNIT OF EACH LOT
0000000001
0000000751
000005001

ENTER LAST UNIT OF EACH LOT
0000000750
000005000
000025000

DATA FOR SLOPE OF 91.00 PCT.

| FIRST UNIT OF LOT | LAST UNIT OF LOT | LOT MID-POINT |
|-------------------|------------------|---------------|
| 1 | 750 | 258 |
| 751 | 5000 | 2527 |
| 5001 | 25000 | 13574 |

MORE DATA FOR MIDPOINT PROGRAM - ANS. YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO
NO

STOP

Subroutine FUNTOT

Problem: (a) What is the First Unit Cost of an item produced on a
learning slope of 87% (-.200913) if the total lot of 6400 items
(first item in lot = 1) cost \$41,600? (b) With the same lot size and cost

FIGURE 5. (CONTINUED)

SAME TOTAL LOT COST - ANS. YES OR NO
YES

ENTER NO. OF FIRST UNIT IN LOT
000001501

ENTER NO. OF LAST UNIT IN LOT
000007900

THE FIRST UNIT COST = \$ 24.72
MORE DATA FOR FIRST UNIT COST - ANS. YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO
NO

Subroutine FUNAV

Problem: (a) What is the First Unit Cost of an item produced on
a learning slope of 87% (-.200913) if the average cost of an item in a
6400 item lot (first item in lot = 1) is \$6.50? (b) With the same lot
size and average unit cost, what is the First Unit Cost if the learning
slope is 89.5% (-.160040), and the first unit in the lot is 7001?

The program output is shown in Figure 6.

FIGURE 6

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.
C

THIS IS FUNAV

ENTER AVERAGE UNIT COST OF LOT
6.50

what is the First Unit Cost if the learning slope is 89.5% (-.160040),
and the first unit in the lot is 1501.

The program output is shown in Figure 5.

FIGURE 5

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

E

THIS IS FUNTOT

ENTER TOTAL COST OF THE LOT
41600.

ENTER LEARNING SLOPE EXPONENT
-.200913

ENTER NO. OF FIRST UNIT IN LOT
000000001

ENTER NO. OF LAST UNIT IN LOT
000006400

THE FIRST UNIT COST = \$ 30.23
MORE DATA FOR FIRST UNIT COST - ANS. YES OR NO
YES

SAME LEARNING SLOPE EXPONENT - ANS. YES OR NO
NO

ENTER LEARNING SLOPE EXPONENT
-.160040

FIGURE 6 (CONTINUED)

ENTER LEARNING SLOPE EXPONENT
-.200913

ENTER NO. OF FIRST UNIT IN LOT
000000001

ENTER NO. OF LAST UNIT IN LOT
000006400

THE FIRST UNIT COST = \$ 30.23
MORE DATA FOR FIRST UNIT COST - ANS. YES OR NO
YES

SAME LEARNING SLOPE EXPONENT - ANS. YES OR NO
NO

ENTER LEARNING SLOPE EXPONENT
-.160040

SAME AVG. UNIT COST - ANS. YES OR NO
YES

ENTER NO. OF FIRST UNIT IN LOT
000007001

ENTER NO. OF LAST UNIT IN LOT
000013400

THE FIRST UNIT COST = \$ 28.38
MORE DATA FOR FIRST UNIT COST - ANS. YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO

NO

STOP

Subroutine TOTCOST/Subroutine WHATIF

The TOTCOST segment sample problem demonstrates how the user may solve several problems dealing with the total cost of a lot, with a single computation. (This short cut can also be accomplished with

Subroutine AVUCOST.) The sample problem for TOTCOST also provides an ideal example for the use of the WHATIF Segment.

Problem: (a) What is the total cost for each of the two lots shown below if the item is produced on a learning slope of 90% (-.152003) and the first unit cost is \$98.50? (b) If the same total quantity is procured in two lots of 1500 and 6000, all else being equal (same first unit cost and learning slope exponent), what will be the cost of each lot? (c) If equal dollar amounts are to be spent for each lot, what lot sizes can be procured?

| | <u>Lot</u> | <u>First Unit</u> | <u>Last Unit</u> |
|-----|------------|-------------------|------------------|
| (a) | 1 | 1 | 2500 |
| | 2 | 2501 | 7500 |
| (b) | 1 | 1 | 1500 |
| | 2 | 1501 | 7500 |

The output for solution to parts (a), (b) above is shown in Figure 7.

FIGURE 7

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

D

THIS IS TOTCOST

ENTER LEARNING SLOPE EXPONENT
-.152003

ENTER FIRST UNIT COST
98.50

FIGURE 7 (CONTINUED)

HOW MANY LOTS

04

ENTER FIRST UNIT OF EACH LOT

000000001
000002501
000000001
000001501

BEST AVAILABLE COPY

ENTER LAST UNIT OF EACH LOT

000002500
000007500
000001500
000007500

DATA FOR:

SLOPE FCT. 90.00
1ST UNIT COST \$ 98.50

| FIRST UNIT OF LOT | LAST UNIT OF LOT | TOTAL COST OF LOT |
|-------------------|------------------|-------------------|
| 1 | 2500 | 88357.09 |
| 2501 | 7500 | 136021.75 |
| 1 | 1500 | 57278.48 |
| 1501 | 7500 | 167100.36 |

Note that the total of the two lots, problem (a) or (b) is \$224,378.84 (as expected). Sample problem (c) seeks to determine what lot sizes could be procured if the total dollar quantity were equally divided, \$112,189.42, between the two lots. The WHATIF Segment provides the solution, Figure 8.

FIGURE 8

MORE DATA FOR TOTAL LOT COST - ANS. YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO
G

THIS IS WHATIF

ENTER NO. OF LOTS (MAX=5)
2

ENTER FIRST UNIT COST
98.50

ENTER LEARNING SLOPE EXPONENT
-.152003

ENTER TOTAL COST OF EACH LOT
112189.42
112189.42

| LOT SIZE | LAST UNIT | LOT COST | AVG. UNIT COST |
|----------|-----------|----------|----------------|
| 3313. | 3313. | 112189. | 33.86 |
| 4187. | 7500. | 112189. | 26.79 |

MORE DATA FOR WHATIF - ANS. YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO
N

STOP

Subroutine MIDBEST

Given a set of historical procurement data the MIDBEST segment
computes the first unit cost and learning slope exponent. The data are

fitted to the logarithmic form of the power curve by classical least mean square techniques. The output also provides a Standard Error of Estimate (Standard Deviation) as a guide to the fit of the input data to the theoretical curve.

The initial query of this segment requests the user to provide a guess of the learning slope. The purpose for this guess is to reduce computation time and the possibility of encountering a computer system time limit. If the user is utterly devoid of any learning exponent estimate he should enter a value of -.234465 (85%). This value of the learning slope is commonly used as a starting point in nonautomated solutions of this problem.

Problem: Four lots of an item were procured in quantities and at average unit costs shown below. Determine (a) the first unit cost of the item and its learning slope.

| <u>Lot</u> | <u>First Item In Lot</u> | <u>Last Item In Lot</u> | <u>Average Unit Cost</u> |
|------------|------------------------------|-----------------------------|------------------------------|
| 1 | 901 | 3632 | \$ 7922 |
| 2 | 3633 | 6958 | 5963 |
| 3 | 6959 | 10860 | 4989 |
| 4 | 10861 | 15624 | 4714 |

Figure 9 is the output for this problem.

FIGURE 9

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

F

THIS IS MIDBEST

FIGURE 9 (CONTINUED)

ENTER YOUR GUESS OF SLOPE EXPONENT (USE NEG SIGN)
-.234465

HOW MANY LOTS
04

ENTER FIRST UNIT OF EACH LOT
000000901
000003633
000006959
000010861

ENTER LAST UNIT OF EACH LOT
000003632
000006958
000010860
000015624

ENTER AVG. UNIT COST OF EACH LOT
7922.
5963.
4989.
4714.

LEARNING SLOPE EXPONENT = -.290403
LEARNING SLOPE PERCENT = 81.77
FIRST UNIT COST = \$ 71997.54
STANDARD ERROR OF ESTIM = \$ 112.84

MORE DATA FOR MIDBEST
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO

NO

STOP

BEST AVAILABLE COPY

Subroutine LOTSIZE

Whereas WHATIF computes the sizes of lots starting from Item 1, the LOTSIZE Segment computes the size of single lot starting anywhere on the cost/cumulative quantity curve. To illustrate the LOTSIZE segment, the sample problem utilizes a part of the data shown in the WHATIF solution, Figure 8.

Problem: A total of \$112,189 is available for procurement of a lot of items which have been produced on a demonstrated learning slope of 90% with a first unit cost of \$98.50. The first unit of this lot will be the 3314th produced. (a) What is the size of lot that can be procured, (b) the final unit of the lot, and, (c) the average unit cost of lot?

Figure 10 is the output for this problem.

FIGURE 10

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

H

THIS IS LOTSIZE

ENTER TOTAL COST OF THE LOT
112189.

ENTER LEARNING SLOPE EXPONENT
-.152003

ENTER FIRST UNIT COST
98.50

ENTER NO. OF FIRST UNIT IN LOT
000003314

BEST AVAILABLE COPY

FIGURE 10 (CONTINUED)

TOTAL LOT SIZE IS 4187.
LAST UNIT IN LOT IS 7500.
AVG. UNIT COST IS 26.79

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO

NO

STOP

Subroutine AVUCOST

Problem: What will be the average cost per unit for each of three lots described below if the items are produced on a learning slope of 80%
(-.321928) and the first unit cost is \$2000.

The program output is shown in Figure 11.

FIGURE 11

FOR SHORT DESCRIPTION OF PROGRAM
TYPE Y OR TYPE SEGMENT LETTER.

E

THIS IS AVUCOST

ENTER LEARNING SLOPE EXPONENT
-.321928

ENTER FIRST UNIT COST
2000.

HOW MANY LOTS
03

BEST AVAILABLE COPY

FIGURE 11 (CONTINUED)

ENTER FIRST UNIT OF EACH LOT

000000901
000001001
000001201

ENTER LAST UNIT OF EACH LOT

000001000
000001200
000001500

DATA FOR:

SLOPE PCT. 80.00
1ST UNIT COST \$ 2000.00

| FIRST UNIT IN LOT | LAST UNIT IN LOT | LOT SIZE | AVG. UNIT COST |
|-------------------|------------------|----------|----------------|
| 901 | 1000 | 100 | 220.00 |
| 1001 | 1200 | 200 | 209.95 |
| 1201 | 1500 | 300 | 196.62 |

MORE DATA FOR AVG. UNIT COST - ANSWER YES OR NO
NO

IF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT
LETTER OR TYPE NO
NO

STOP

*BEST AVAILABLE COPY

PROGRAM LISTING

```

PROGRAM XPCURVE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPJ1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
DATA LOTF,LOTL,IZ,HZ/60*//
DATA YEA,Y,HY,YE,ZI,AC,X/105*0/
DATA AD,DF,DL,LD,XN,T,SL/35*//
PRINT 25
25 FORMAT(3X,32HFOR SHORT DESCRIPTION OF PROGRAM,/,
+3X,3)HTYPE Y OR TYPE SEGMENT LETTER.,/)
READ 69, QC
69 FORMAT(A2)
IF(QC.EQ.2HY)GOTO 113
IF(QC.EQ.2HA)CALL MIDPINT,RETURNS(52)
IF(QC.EQ.2HB)CALL FUNTOT,RETURNS(52)
IF(QC.EQ.2HC)CALL FUNAV,RETURNS(52)
IF(QC.EQ.2HD)CALL TOTCOST,RETURNS(52)
IF(QC.EQ.2HE)CALL AVUCOST,RETURNS(52)
IF(QC.EQ.2HF)CALL MIDBEST,RETURNS(52)
IF(QC.EQ.2HG)CALL WHATIF,RETURNS(52)
IF(QC.EQ.2HH)CALL LOTSIZE,RETURNS(52)
113 PRINT 81
81 FORMAT(3X,48HTHIS PROGRAM CONSISTS OF THE FOLLOWING SEGMENTS//,
+8X,38HA. MIDPOINT - DETERMINES LOT MIDPOINTS/
+8X,40HB. FUNTOT - DETERMINES FIRST UNIT COST/
+8X,40HC. FUNAV - DETERMINES FIRST UNIT COST/
+8X,42HD. TOTCOST - DETERMINES TOTAL COST OF LOT/
+8X,+2HE. AVUCOST - DETERMINES ITEM AVERAGE COST/
+8X,40HF. MIDBEST - DETERMINES FIRST UNIT COST/
+22X,27HAND LEARNING SLOPE EXPONENT/
+8X,*G. WHATIF - DETERMINES LOT SIZES*/
+8X,*H. LOTSIZE - DETERMINES SIZE OF A SINGLE LOT*///
+3X,+6HIF YOU NEED MORE DETAIL TYPE LETTER OR TYPE NO./)
82 READ 12,Q
12 FORMAT(A2)
ICOUNT=ICOUNT+1
IF(ICOUNT.EQ.9)GOTO 44
IF(Q.EQ.2HA)GOTO 17
IF(Q.EQ.2HB)GOTO 23
IF(Q.EQ.2HC)GOTO 26
IF(Q.EQ.2HD)GOTO 29
IF(Q.EQ.2HE)GOTO 31
IF(Q.EQ.2HF)GOTO 34
IF(Q.EQ.2HG)GOTO 88
IF(Q.EQ.2HH)GOTO 77
PRINT 13
13 FORMAT(3X,36HTYPE IN LETTER OF PROGRAM YOU DESIRE,/)

10 READ 14,Q1
14 FORMAT(A1)
IF(Q1.EQ.1HA)CALL MIDPINT,RETURNS(52)
IF(Q1.EQ.1HB)CALL FUNTOT,RETURNS(52)
IF(Q1.EQ.1HC)CALL FUNAV,RETURNS(52)
IF(Q1.EQ.1HD)CALL TOTCOST,RETURNS(52)

```

```

IF(Q1.EQ.1HE)CALL AVUCOST,RETURNS(52)
IF(Q1.EQ.1HF)CALL MIDBEST,RETURNS(52)
IF(Q1.EQ.1HG)CALL WHATIF,RETURNS(52)
IF(Q1.EQ.1HH)CALL LOTSIZE,RETURNS(52)
PRINT 15
15 FORMAT (3X,24HWRONG LETTER - TRY AGAIN,/)

GOTO 16
17 PRINT 18
18 FORMAT (3X,16HPROGRAM MIDPOINT//  

+7X,53HTHIS SEGMENT CALCULATES LOT MIDPOINTS IF THE LEARNING/  

+7X,56HSLOPE EXPONENT AND LOT QUANTITIES (FIRST AND LAST ITEMS)/  

+7X,44HARE KNOWN. TYPE A TO CALL PROGRAM OR TYPE NO,/)  

READ 19,Q2
19 FORMAT (A2)
IF(Q2.EQ.2HA )CALL MIDPINT,RETURNS(52)
20 PRINT 21
21 FORMAT (3X,46HWHAT PROGRAM DO YOU WANT DESCRIBED-TYPE LETTER,/)  

GOTO 82
23 PRINT 24
24 FORMAT (3X,14HPROGRAM FUNTOT//  

+7X,55HTHIS SEGMENT CALCULATES FIRST UNIT COST IF THE LEARNING/  

+7X,53HSLOPE EXPONENT, LOT QUANTITIES (FIRST AND LAST ITEMS)/  

+7X,56HAND TOTAL COST OF THE LOT ARE KNOWN. TYPE B TO CALL THIS/  

+7X,18HPROGRAM OR TYPE NO,/)  

READ 19,Q3
IF(Q3.EQ.2HB )CALL FUNTOT,RETURNS(52)
GOTO 20
26 PRINT 27
27 FORMAT (3X,13HPROGRAM FUNAV//  

+7X,55HTHIS SEGMENT CALCULATES FIRST UNIT COST IF THE LEARNING/  

+7X,53HSLOPE EXPONENT, LOT QUANTITIES (FIRST AND LAST ITEMS)/  

+7X,58HAND THE AVERAGE UNIT COST OF AN ITEM IN THE LOT ARE KNOWN./  

+7X,38HTYPE C TO CALL THIS PROGRAM OR TYPE NO,/)  

READ 19,Q4
IF(Q4.EQ.2HC )CALL FUNAV,RETURNS(52)
GOTO 20
29 PRINT 30
30 FORMAT (3X,15HPROGRAM TOTCOST//  

+7X,54HTHIS SEGMENT CALCULATES THE TOTAL COST OF A LOT IF THE/  

+7X,55HLEARNING SLOPE EXPONENT, FIRST UNIT COST, AND LOT QUAN-/  

+7X,56HTIES (FIRST AND LAST ITEMS), ARE KNOWN. TYPE D TO CALL/  

+7X,23HTHIS PROGRAM OR TYPE NO,/)  

READ 19,Q5
IF(Q5.EQ.2HD )CALL TOTCOST,RETURNS(52)
GOTO 20
31 PRINT 32
32 FORMAT (3X,15HPROGRAM AVUCOST//  

+7X,56HTHIS SEGMENT CALCULATES THE AVERAGE COST OF AN ITEM IN A/  

+7X,55HLOT IF THE LEARNING SLOPE EXPONENT, FIRST UNIT COST AND/  

+7X,56HLOT QUANTITIES (FIRST AND LAST ITEMS), ARE KNOWN. TYPE E/  

+7X,31HTO CALL THIS PROGRAM OR TYPE NO,/)  

READ 19,Q6
IF(Q6.EQ.2HE )CALL AVUCOST,RETURNS(52)

```

```

        GOTO 20
34 PRINT 35
35 FORMAT (3X,15HPROGRAM MIDBEST//  

+7X,58HTHIS SEGMENT CALCULATES FIRST UNIT COST AND LEARNING SLOPE/  

+7X,58HEXPONENT FROM LOT DATA IF THE AVERAGE UNIT COST OF AN ITEM/  

+7X,54HIN EACH OF THE LOTS AND LOT QUANTITIES (FIRST AND LAST/  

+7X,58HITEMS) ARE KNOWN. LEAST MEAN SQUARE FIT OF DATA IS UTILIZ-/  

+7X,58HED. STANDARD ERROR OF ESTIMATE AND CORRELATION COEFFICIENT/  

+7X,57HARE CALCULATED. PROGRAM CAN ACCOMODATE MAXIMUM OF NINE(9)/  

+7X,45HLLOTS. TYPE F TO CALL THIS PROGRAM OR TYPE NO./)  

READ 19,Q7  

IF(Q7.EQ.2HF )CALL MIDBEST,RETURNS(52)
GOTO 20
88 PRINT 240
240 FORMAT(3X,*PROGRAM WHATIF*//,  

+7X,*THIS SEGMENT COMPUTES THE SIZE OF 1 OR MORE LOTS(5 MAX), IF*/  

+7X,*THE TOTAL COST OF THE LOT(S), THE 1ST UNIT COST, AND THE*/  

+7X,*LEARNING SLOPE EXPONENT ARE KNOWN. TYPE G TO CALL THIS PRO-*/  

+7X,*GRAM OR TYPE NO*,/)  

READ 19,Q9  

IF(Q9.EQ.2HG )CALL WHATIF,RETURNS(52)
GOTO 20
77 PRINT 66
66 FORMAT(3X,*PROGRAM LOTSIZE*//  

+7X,*THIS SEGMENT COMPUTES THE SIZE OF A SINGLE LOT IF THE TOTAL*/  

+7X,*COST OF THE LOT, THE 1ST UNIT COST, THE LEARNING SLOPE EX-*/  

+7X,*PONENT AND THE THE FIRST UNIT OF THE LOT ARE KNOWN. IT DIF-*/  

+7X,*FERS FROM SEGMENT WHATIF IN THAT THE FIRST UNIT OF LOTSIZE*/  

+7X,*CAN BE ANY UNIT (OTHER THAN UNIT NO. 1). TYPE H TO CALL*/  

+7X,*THIS PROGRAM OR TYPE NO*)  

READ 19,Q10
IF(Q10.EQ.2HH )CALL LOTSIZE,RETURNS(52)
GOTO 20
52 PRINT 37
37 FORMAT (3X,43HIF YOU DESIRE ADDITIONAL PROGRAMS TYPE PERTINENT/  

+3X,17HLETTER OR TYPE NO,/)  

22 READ 19,Q8
IF(Q8.EQ.2HA )CALL MIDPINT,RETURNS(52)
IF(Q8.EQ.2HB )CALL FUNTOT,RETURNS(52)
IF(Q8.EQ.2HC )CALL FUNAV,RETURNS(52)
IF(Q8.EQ.2HD )CALL TOTCOST,RETURNS(52)
IF(Q8.EQ.2HE )CALL AVUCOST,RETURNS(52)
IF(Q8.EQ.2HF )CALL MIDBEST,RETURNS(52)
IF(Q8.EQ.2HG )CALL WHATIF,RETURNS(52)
IF(Q8.EQ.2HH )CALL LOTSIZE,RETURNS(52)
GOTO 1
44 PRINT 33
33 FORMAT(* YOU HAVE EXHAUSTED ALL DESCRIPTION. IF YOU DO NOT NOW*/  

+* TYPE A PERTINENT LETTER FOR A PROGRAM SEGMENT THE PROGRAM WILL*/  

+* BE TERMINATED.*/)  

GOTO 22
1 STOP
END

```

```

SUBROUTINE AVUCOST,RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
PRINT 10
1. FORMAT (3X,15HTHIS IS AVUCOST,////)
PRINT 132
132 FORMAT(3X,* ENTER LEARNING SLOPE EXPONENT*)
READ 62,B
62 FORMAT (F8.6)
PRINT 64
64 FORMAT (3X,21HENTER FIRST UNIT COST,/)
READ 65,A
65 FORMAT (F10.2)
63 PRINT 54
54 FORMAT(3X,* HOW MANY LOTS*)
READ 148, NA
148 FORMAT (I2)
IA=1
PRINT 150
150 FORMAT (3X,28HENTER FIRST UNIT OF EACH LOT,/)
151 READ 152, N2
152 FORMAT (I9)
LOTF(IA)=N2
IA=IA+1
IF(IA.LE.NA)GOTO 151
IB=1
PRINT 153
153 FORMAT (3X,27HENTER LAST UNIT OF EACH LOT,/)
154 READ 152, N1
LOTL(IB)=N1
IB=IB+1
IF(IB.LE.NA)GOTO 154
550 IC=1
155 C=FLOAT(LOTL(IC))+.5
D=FLOAT(LOTF(IC))- .5
E=C-D
DO 80 J=1,NA
80 LOTS(J)=LOTL(J)-LOTF(J)+1
F=1.+B
HY(IC)=A*(C**F-D**F)/(E**F)
IC=IC+1
IF(IC.LE.NA)GOTO 155
PRINT 156
156 FORMAT(1X,1H ,/)
Z5=100.*10.**(-B* ALOG10(0.5))
PRINT 43, Z5, A
43 FORMAT (5X,9HDATA FOR/,/5X,19HSLOPE PCT. ,F5.2/
+5X,15H1ST UNIT COST $,F9.2//)
PRINT 157
157 FORMAT(* FIRST UNIT IN LOT LAST UNIT IN LOT*,
+* LOT SIZE AVG. UNIT COST*//)
PRINT 158,(LOTF(ID),LOTL(ID),LOTS(ID),HY(ID),ID=1,NA)

```

```
158 FORMAT(4X,I9,10X,I9,6X,I9,F14.2)
PRINT 212
212 FORMAT (1X,1H ,///)
PRINT 71
71 FORMAT (3X,45HMORE DATA FOR AVG. UNIT COST - ANS. YES OR NO,/)
READ 72,Z1
72 FORMAT (A3)
IF(Z1.EQ.3HNO) GOTO 52
PRINT 73
73 FORMAT (3X,45HSAME LEARNING SLOPE EXPONENT - ANS. YES OR NO,/)
READ 72,Z2
IF(Z2.EQ.3HNO) GOTO 79
230 PRINT 74
74 FORMAT (3X,37HSAME FIRST UNIT COST - ANS. YES OR NO,/)
READ 72,Z3
IF(Z3.EQ.3HNO) GOTO 67
GOTO 549
79 PRINT 132
READ 62, B
GOTO 230
67 PRINT 70
70 FORMAT (3X,25HENTER NEW FIRST UNIT COST,/)
READ 65, A
549 PRINT 551
551 FORMAT(* SAME NO. OF LOTS AND LOT SIZES-ANS. YES OR NO*/)
READ 72,Z4
IF(Z4.EQ.3HYES) GOTO 550
GOTO 63
52 RETURN A1
END
```

BEST AVAILABLE COPY

```

SUBROUTINE FUNAV,RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/A0(5),OF(5),OL(5),LD(5),XN(5),T(5),SL(5)
PRINT 6
6 FORMAT (3X,13HTHIS IS FUNAV,////)
PRINT 10
100 FORMAT (3X,3HENTER AVERAGE UNIT COST OF LOT,/)
READ 102, V
102 FORMAT (F10.2)
PRINT 132
132 FORMAT (3X,* ENTER LEARNING SLOPE EXPONENT*/)
READ 104,B
104 FORMAT (F10.6)
115 PRINT 105
105 FORMAT (3X,3HENTER NO. OF FIRST UNIT IN LOT,/)
READ 106, N2
106 FORMAT (I9)
PRINT 107
107 FORMAT (3X,29HENTER NO. OF LAST UNIT IN LOT,/)
READ 106, N1
D = FLOAT(N2) - 0.5
C = FLCAT(N1) + 0.5
E = C - D
F = 1.0 + B
A = E*V*F/(C**F - D**F)
PRINT 108,A
108 FORMAT (28X,23HTHE FIRST UNIT COST = $,F10.2)
PRINT 101
101 FORMAT (3X,46HMORE DATA FOR FIRST UNIT COST - ANS. YES OR NO,/)
READ 98,21
98 FORMAT (A3)
IF(Z1.EQ.3HNO )GOTO 52
PRINT 112
112 FORMAT (3X,45HSAME LEARNING SLOPE EXPONENT - ANS. YES OR NO,/)
READ 98,22
IF(Z2.EQ.3HNO )GOTO 60
78 PRINT 221
221 FORMAT (3X,36HSAME AVG. UNIT COST - ANS. YES OR NO,/)
READ 98,23
IF(Z3.EQ.3HYES)GOTO 115
PRINT 128
128 FORMAT (3X,24HENTER NEW AVG. UNIT COST,/)
READ 102, V
GOTO 115
60 PRINT 132
READ 104, B
GOTO 79
52 RETURN A1
END

```

```

SUBROUTINE FUNTOT,RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
PRINT 4
4 FORMAT (3X,14HTHIS IS FUNTOT,////)
PRINT 20
20 FORMAT (3X,27HENTER TOTAL COST OF THE LOT,/)
READ 202, U
202 FORMAT (F10.2)
PRINT 132
132 FORMAT (3X,* ENTER LEARNING SLOPE EXPONENT*)*
READ 204,B
204 FORMAT (F10.6)
215 PRINT 215
205 FORMAT (3X,3HENTER NO. OF FIRST UNIT IN LOT,/)
READ 206, N2
206 FORMAT (I9)
PRINT 207
207 FORMAT (3X,29HENTER NO. OF LAST UNIT IN LOT,/)
READ 206, N1
D = FLOAT(N2) - 0.5
C = FLOAT(N1) + 0.5
F = 1.0 + B
A = U*F/(C**F - D**F)
PRINT 208,A
208 FORMAT (28X,23HTHE FIRST UNIT COST = $,F10.2)
PRINT 188
188 FORMAT (3X,46HMORE DATA FOR FIRST UNIT COST - ANS. YES OR NO,/)
READ 91,Z1
91 FORMAT (A3)
IF(Z1.EQ.3HNO) GOTO 52
PRINT 121
121 FORMAT (3X,45HSAME LEARNING SLOPE EXPONENT - ANS. YES OR NO,/)
READ 91,Z2
IF(Z2.EQ.3HNO) GOTO 47
51 PRINT 232
232 FORMAT (3X,36HSAME TOTAL LOT COST - ANS. YES OR NO,/)
READ 91,Z3
IF(Z3.EQ.3HYES) GOTO 215
PRINT 53
53 FORMAT (3X,24HENTER NEW TOTAL LOT COST,/)
READ 202, U
GOTO 215
47 PRINT 132
READ 204, B
GOTO 51
52 RETURN A1
END

```

```

SUBROUTINE LOTSIZE,RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
PRINT 130
130 FORMAT(* THIS IS LOTSIZE*)
PRINT 200
200 FORMAT(3X,* ENTER TOTAL COST OF THE LOT*)
READ 202,U
202 FORMAT(F10.2)
PRINT 132
132 FORMAT(3X,* ENTER LEARNING SLOPE EXPONENT*)
READ 204,B
204 FORMAT(F10.6)
PRINT 164
164 FORMAT(3X,*ENTER FIRST UNIT COST*)
READ 165,A
165 FORMAT(F8.2)
PRINT 105
105 FORMAT(3X,*ENTER NO. OF FIRST UNIT IN LOT*)
READ 106,N2
106 FORMAT(I9)
D=FLOAT(N2)-.5
F=1.+B
CCD=(F*U/(A)+D**F)**(1/F)-.5
TLS=CCD+.5-D
XAC=U/TLS
PRINT 131,TLS,CCD,XAC
131 FORMAT(5X,*TOTAL LOT SIZE IS*,F12.0,
+/5X,*LAST UNIT IN LOT IS*,F10.0,
+/5X,*AVG. UNIT COST IS*,F11.2///)
RETURN A1
END

```

```

SUBROUTINE MDOBEST, RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
DIMENSION YC(9)
PRINT 36
36 FORMAT (3X,15HTHIS IS MDOBEST,////)
PRINT 38
38 FORMAT(* ENTER YOUR GUESS OF SLOPE EXPONENT (USE NEG SIGN)*/)
READ 39,B
39 FORMAT(F9.6)
211 PRINT 54
54 FORMAT(3X,* HOW MANY LOTS*)
READ 180, N
180 FORMAT (I2)
IA=1
PRINT 181
181 FORMAT (3X,28HENTER FIRST UNIT OF EACH LOT,/)
182 READ 183, K
183 FORMAT (I9)
LOTF(IA)=K
IA=IA+1
IF(IA.LE.N)GOTO 182
IB=1
PRINT 184
184 FORMAT (3X,27HENTER LAST UNIT OF EACH LOT,/)
185 READ 183, L
LOTL(IB) = L
IB=IB+1
IF(IB.LE.N)GOTO 185
160 ID=1
PRINT 186
186 FORMAT (3X,32HENTER AVG. UNIT COST OF EACH LOT,/)
187 READ 87, AUC
87 FORMAT (F10.2)
AC(ID)=AUC
Y(ID)= ALOG10(AUC)
ID=ID+1
IF(ID.LE.N)GOTO 187
189 IC=1
Z=100.*10.**(-B*ALOG10(0.5))
190 C=FLOAT(LOTL(IC))+.5
D=FLOAT(LOTF(IC))-.
E=C-D
F=1.0+B
H=-1.0/B
POINT=((E*F)/(C**F-D**F))**H
IZA=IFIX(POINT + .5)
ZI(IC)=FLOAT(IZA)
X(IC)=ALOG10(ZI(IC))
IC=IC+1
IF(IC.LE.N)GOTO 190
SUMX = 0.0

```

```

SUMY = 0.0
SUMXY = 0.0
SUMXSQ = 0.0
I = 1
191 SUMX=SUMX+X(I)
      SUMY = SUMY + Y(I)
      SUMXY = SUMXY + X(I)*Y(I)
      SUMXSQ = SUMXSQ + X(I)**2.
      I = I + 1
      IF(I.LE.N)GOTO 191
      SUMSQX = SUMX**2.
      EN = FLOAT(N)
      F = (SUMY*SUMXSQ - SUMX*SUMXY)/(EN*SUMXSQ - SUMSQX)
      G = (EN*SUMXY - SUMX*SUMY)/(EN*SUMXSQ - SUMSQX)
      FIRSTU = 10.**F
      SPCT=100.*10.**(-G*ALOG1(0.5))
      IF(ABS(Z-SPCT).GT.0.5)B=G
      IF(B.EQ.G)GOTO 189
      PRINT 192, G,SPCT,FIRSTU
192 FORMAT (25X,26HLEARNING SLOPE EXPONENT = ,F12.6/
+25X,26HLEARNING SLOPE PERCENT = ,6X,F6.2/
+25X,30HFIRST UNIT COST = $ ,F8.2)
      STD=0.
      DO 2 JQ=1,N
      YC(JQ)=F+G*X(JQ)
2      STD=STD+(10.**YC(JQ)-AC(JQ))**2.
      SEE=SQRT(STD/FLOAT(N))
      PRINT 1,SEE
1      FORMAT(25X,*STANDARD ERROR OF ESTIM = $*,F11.2)
      PRINT 201
201 FORMAT (3X,1H ,///)
      PRINT 209
209 FORMAT (3X,21HMORE DATA FOR MIDBEST,/)

      READ 210, QA
210 FORMAT (A3)
      IF(QA.EQ.3HNO )GOTO 52
      PRINT 551
551 FORMAT(* SAME NO. OF LOTS AND LOT SIZES-ANS. YES OR NO*)
      READ 72,Z8
72  FORMAT(A3)
      IF(Z8.EQ.3HYES)GOTO 160
      GOTO 211
52  RETURN A1
END

```

BEST AVAILABLE COPY

```

SUBROUTINE MIDPOINT,RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YEA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),TI(15),AC(15),IZ(15)
COMMON/C/AD(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
PRINT 2
2 FORMAT (3X,16HTHIS IS MIDPOINT,///)
PRINT 132
132 FORMAT(3X,29HENTER LEARNING SLOPE EXPONENT,/1
READ 133, B
133 FORMAT (F8.6)
159 PRINT 54
54 FORMAT (3X,13HHOW MANY LOTS,/1
READ 45, NA
45 FORMAT (I2)
IA=1
PRINT 46
46 FORMAT (3X,28HENTER FIRST UNIT OF EACH LOT,/1
117 READ 118, N2
118 FORMAT (I9)
LOTF(IA)=N2
IA=IA+1
IF(IA.LE.NA)GOTO 117
IB=1
PRINT 119
119 FORMAT (3X,27HENTER LAST UNIT OF EACH LOT,/1
120 READ 118, N1
LOTL(IB)=N1
IB=IB+1
IF(IB.LE.NA)GOTO 121
IC=1
122 C=FLOAT(LOTL(IC))+.5
D=FLOAT(LOTF(IC))- .5
E = C - D
F = 1.0 + B
H = -1.0/B
Z = ((E*F)/(C**F - D**F))**H
IZ(IC)=IFIX(Z+.5)
IC=IC+1
IF(IC.LE.NA)GOTO 122
PRINT 126
126 FORMAT (3X,1H ,///)
Z6=100.*10.**(-B* ALOG10(0.5))
PRINT 76, Z6
76 FORMAT (3CX,18HDATA FOR SLOPE OF ,F5.2,5H PCT.,///)
PRINT 124
124 FORMAT (4X,17HFIRST UNIT OF LOT,4X,16HLAST UNIT OF LOT,4X,
+13HLOT MID-POINT,/1
PRINT 125, (LOTF(ID),LOTL(ID),IZ(ID), ID=1,NA)
125 FORMAT (9X,I9,11X,I9,9X,I9)
PRINT 127
127 FORMAT (3X,1H ,///)
PRINT 48
48 FORMAT (3X,47HMORE DATA FOR MIDPOINT PROGRAM - ANS. YES OR NO,/1

```

```
READ 49, Q
49 FORMAT (A3)
IF(Q.EQ.'HNO')GOTO 52
PRINT 51
51 FORMAT (3X,45H SAME LEARNING SLOPE EXPONENT - ANS. YES OR NO,/1)
READ 49,R
IF(R.EQ.'HNO')GOTO 213
GOTO 159
213 PRINT 132
READ 42,B
42 FORMAT (F8.6)
GOTO 159
52 RETURN A1
END
```

BEST AVAILABLE COPY

```

SUBROUTINE TOTCOST, RETURNS(A1)
COMMON/A/LOTF(15), LOTL(15), LOTS(15), YEA(15), X(15), Y(15)
COMMON/B/HZ(15), HY(15), YE(15), ZI(15), AC(15), IZ(15)
COMMON/C/AD(5), DF(5), DL(5), LD(5), XN(5), T(5), SL(5)
PRINT 8
8 FORMAT (3X,15HTHIS IS TOTCOST,////)
PRINT 132
132 FORMAT (3X,* ENTER LEARNING SLOPE EXPONENT*)
READ 162, B
162 FORMAT (F11.6)
PRINT 164
164 FORMAT (3X,21HENTER FIRST UNIT COST,/)
READ 165, A
165 FORMAT (F8.2)
134 PRINT 54
54 FORMAT (3X,* HOW MANY LOTS*)
READ 136, NA
136 FORMAT (I2)
IA=1
PRINT 167
167 FORMAT (3X,28HENTER FIRST UNIT OF EACH LOT,/)
137 READ 168, N2
168 FORMAT (I9)
LOTF(IA)=N2
IA=IA+1
IF(IA.LE.NA)GOTO 137
IB=1
PRINT 169
169 FORMAT (3X,27HENTER LAST UNIT OF EACH LOT,/)
138 READ 168, N1
LOTL(IB)=N1
IB=IB+1
IF(IB.LE.NA)GOTO 138
IC=1
139 C=FLOAT(LOTL(IC))+.5
D=FLOAT(LOTF(IC))- .5
F=1.+B
HZ(IC)=A*(C**F-D**F)/F
IC=IC+1
IF(IC.LE.NA)GOTO 139
PRINT 140
140 FORMAT (1X,1H ,////)
Z7=100.*10.**(-B*ALOG10(.5))
PRINT 123, Z7, A
129 FORMAT (5X,9HDATA FOR:, /5X,19HSLOPE PCT. ,F5.2/
+5X,15H1ST UNIT COST $,F9.2//)
PRINT 171
171 FORMAT (4X,17HFIRST UNIT OF LOT,4X,16HLAST UNIT OF LOT,4X,17HTOTAL
+ COST OF LOT,/)
PRINT 141, (LOTF(ID),LOTL(ID),HZ(ID), ID=1,NA)
141 FORMAT (9X,I9,11X,I9,9X,F12.2)
PRINT 163
163 FORMAT (1X,1H ,////)

```

```
PRINT 171
171 FORMAT (3X,45HMORE DATA FOR TOTAL LOT COST - ANS. YES OR NO,/)
READ 172,Z1
172 FORMAT (A3)
IF(Z1.EQ.3HNO )GOTO 52
PRINT 173
173 FORMAT (3X,45HSAME LEARNING SLOPE EXPONENT - ANS. YES OR NO,/)
READ 172,Z2
IF(Z2.EQ.3HNO )GOTO 143
142 PRINT 17+
174 FORMAT (3X,37HSAME FIRST UNIT COST - ANS. YES OR NO,/)
READ 172,Z3
IF(Z3.EQ.3HNO )GOTO 145
GOTO 134
143 PRINT 132
READ 162, B
GOTO 142
145 PRINT 140
146 FORMAT (3X,25HENTER NEW FIRST UNIT COST,/ )
READ 165, A
GOTO 134
52 RETURN A1
END
```

BEST AVAILABLE COPY

```

SUBROUTINE WHATIF, RETURNS(A1)
COMMON/A/LOTF(15),LOTL(15),LOTS(15),YFA(15),X(15),Y(15)
COMMON/B/HZ(15),HY(15),YE(15),ZI(15),AC(15),IZ(15)
COMMON/C/AJ(5),DF(5),DL(5),LD(5),XN(5),T(5),SL(5)
PRINT 224
224 FORMAT(3X,*THIS IS WHATIF*////)
238 PRINT 239
239 FORMAT(3X,*ENTER NO. OF LOTS (MAX=5)*/)
READ 241,N
241 FORMAT(I1)
PRINT 6-
64 FORMAT(3X,*ENTER FIRST UNIT COST*/)
READ 65,A
65 FORMAT(F10.2)
PRINT 132
132 FORMAT(3X,* ENTER LEARNING SLOPE EXPONENT*)
READ 62,B
62 FORMAT(F8.0)
PRINT 11
11 FORMAT(3X,*ENTER TOTAL COST OF EACH LOT*,/)
DO 227 IX=1,N
227 READ 212,T(IX)
212 FORMAT(F10.2)
K=1
F=1.+B
DF(1)=1.
231 XN(K)=(F*T(K)/A)+(DF(K)-.5)**F
LD(K)=IFIX((XN(K)**(1./F)))+.5
DL(K)=FLOAT(LD(K))
SL(K)=DL(K)-DF(K)+1.
AD(K)=T(K)/SL(K)
K=K+1
IF(K.GT.N)GOTO 234
DF(K)=DL(K-1)+1.
GOTO 231
234 PRINT 235
235 FORMAT(/5X,*LOT SIZE*,4X,*LAST UNIT*,7X,
+*LOT COST*,4X,*AVG. UNIT COST*,/)
PRINT 236,(SL(JK),DL(JK),T(JK),AD(JK),JK=1,N)
236 FORMAT(3X,F11.0,2X,F11.0,F15.0,6X,F9.2)
PRINT 237
237 FORMAT(//* MORE DATA FOR WHATIF - ANS. YES OR NO*/)
READ 72,Q10
72 FORMAT(A3)
IF(Q10.EQ.3HYES)GOTO 238
RETURN A1
END

```

BEST AVAILABLE COPY

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|-----------------------|---|
| 1. REPORT NUMBER PMSA-2-6 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle) A COMPUTER PROGRAM FOR SOLVING LEARNING CURVE PROBLEMS | | 5. TYPE OF REPORT & PERIOD COVERED FINAL |
| 7. AUTHOR(s) LOUIS M. SMITH | | 6. PERFORMING ORG. REPORT NUMBER |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS DARCOM OFFICE OF THE PROJECT MANAGER FOR SELECTED AMMUNITION, DOVER, N.J. | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS | | 12. REPORT DATE DECEMBER 1977 |
| 14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) | | 13. NUMBER OF PAGES 37 |
| | | 15. SECURITY CLASS. (of this report) UNCLASSIFIED |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |
| 16. DISTRIBUTION STATEMENT (of this Report) Approved for public release. Distribution unlimited. | | |
| 17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) | | |
| 18. SUPPLEMENTARY NOTES | | |
| 19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Learning curve, problem solving, computer model. | | |
| 20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Report describes a computer program for solving Learning Curve problems. Program is written in Fortran IV, and is terminal/user interactive. Typical problems solved are lot Mid-Points, First Unit Costs, Total and specific lot costs, learning slopes and first unit cost from raw lot data, and a what-if segment that predicts lot sizes. The program also provides some instruction for the user to enable proper usage. | | |

